A STOCHASTIC PROGRAMMING APPROACH FOR CHEMOTHERAPY APPOINTMENT SCHEDULING

Nur Banu Demir[1], Serhat Gul[2], Melih Çelik[3]

[1]Department of Industrial and Systems Engineering
Wayne State University, Detroit, Michigan, USA
[2]Department of Industrial Engineering, TED University, Ankara, Turkey
[3]School of Management, University of Bath, Bath, UK

Correspondence
Serhat Gul, Department of Industrial Engineering
TED University, 06420, Çankaya, Ankara, Turkey
E-mail: serhat.gul@tedu.edu.tr

September 2020

A Stochastic Programming Approach for Chemotherapy Appointment Scheduling

September 16, 2020

Abstract

Chemotherapy appointment scheduling is a challenging problem due to the uncertainty in pre-medication and infusion durations. In this paper, we formulate a two-stage stochastic mixed integer programming model for the chemotherapy appointment scheduling problem under limited availability of nurses and infusion chairs. The objective is to minimize the expected weighted sum of nurse overtime, chair idle time, and patient waiting time. The computational burden to solve real-life instances of this problem to optimality is significantly high, even in the deterministic case. To overcome this burden, we incorporate valid bounds and symmetry breaking constraints. Progressive hedging algorithm is implemented in order to solve the improved formulation heuristically. We enhance the algorithm through a penalty update method, cycle detection and variable fixing mechanisms, and a linear approximation of the objective function. Using numerical experiments based on real data from a major oncology hospital, we compare our solution approach with several scheduling heuristics from the relevant literature, generate managerial insights related to the impact of the number of nurses and chairs on appointment schedules, and estimate the value of stochastic solution to assess the significance of considering uncertainty.

Keywords: chemotherapy, scheduling, appointment scheduling, stochastic programming, progressive hedging

1 Introduction

Recent statistical data on global health shows that cancer is one of the most prevalent causes of death, second only to heart disease (Ritchie and Roser, 2018). The number of new cancer cases per year are estimated to increase from 17 million to 23.6 million by 2030 (Cancer Research UK, 2018; National Cancer Institute, 2018). In 2017, estimated expenditures for cancer treatment were $147.3 billion in the US (National Cancer Institute, 2018). Since many people are affected by this disease, it is crucial to have planning systems which decrease costs and increase patient satisfaction at the same time.

Chemotherapy is a frequently used method to cure cancer patients. Chemotherapy treatment is an exhausting process for patients, since side effects may be observed during or after the treatment. To reduce the intensity of the side effects, pre-medication drugs are injected to the patients before initializing the treatment. After the pre-medication process, the patients receive chemotherapy drugs. These drugs should be given to patients based on predetermined frequency and doses.

Designing patient schedules is vitally important due to the limited capacity of oncology clinics and time restrictions for the chemotherapy treatment. Oncology clinics should provide schedules that consider the trade-off between provider and patient satisfaction. In the literature and in practice, chemotherapy schedules are generally created in two phases: (i) *chemotherapy planning*, where patients are assigned to days according to their treatment frequency and (ii) *chemotherapy scheduling*, where patients are sequenced and their appointment times are set to create a schedule for a given day. In this study, we concentrate on constructing daily schedules for chemotherapy patients.

Nurse overtime, chair idle time and patient waiting time are the three criteria that should be considered to evaluate the effectiveness of chemotherapy scheduling. Nurse overtime and chair idle time are undesirable for providers, since they increase operating costs of the clinic. Furthermore, working for more than the shift length may increase nurse dissatisfaction, and which in turn may lead to higher nurse turnover rates. Reducing patient waiting time is also important for clinics to improve patient satisfaction and service level. Since treatment durations are not actually known in advance of the treatment, rough estimates of these durations are generally used for scheduling. However, using such simple estimates may result in an undesirable amount of waiting time, idle time and overtime, particularly when the estimates are not very accurate.

The most complicating factor for chemotherapy patient appointment scheduling is the uncer-

tainty in treatment durations. The treatment durations are unknown due to reasons including a change in the prescription list according to patient's health status before initiating the treatment, complications in patients, and early treatment termination due to the patient not tolerating the treatment (Castaing et al., 2016). If the decision maker plans the daily schedule based on the longest possible treatment durations, patient waiting time can be reduced. However, this may lead to excessive nurse overtime and chair idle time. Similarly, schedules based on the shortest possible treatment durations may result in high waiting times in the clinic. Therefore, the decision maker should handle the uncertainty in treatment durations wisely to avoid long waiting time, chair idle time and nurse overtime.

The limited availability of chairs and nurses in the system is another crucial factor in this problem. A patient should simultaneously seize a nurse and an infusion chair for the treatment. The pre-medication process involves patients receiving pre-medication drugs, if required in the prescription. A nurse can conduct at most one pre-medication at a time. During chemotherapy infusion, the drugs that are used in chemotherapy treatment are injected to patients, in which case a single nurse proctors multiple patients.

In this paper, we study the problem of sequencing patients and setting appointment times (i.e., scheduling appointment start times) for a chemotherapy unit considering the availability of a limited number of nurses and chairs under pre-medication and infusion duration uncertainties. Decisions in this problem include (1) sequencing patients of a daily appointment list, (2) setting appointment times, (3) assignment of patients to nurses, and (4) assignment of patients to chairs. We model the problem using a two-stage stochastic mixed integer programming (SMIP) formulation. We consider an objective function that minimizes the total expected penalty of patient waiting time, chair idle time and nurse overtime across a large set of scenarios sampled through pre-medication and infusion time distributions. We propose a modified progressive hedging algorithm (PHA) that utilizes the problem structure to find near-optimal chemotherapy schedules. In particular, we propose a penalty update method that considers convergence behavior of the primal and dual variables. The method includes a limit on the penalty parameter, whose value changes according to the iteration number. We also make use of cycle detection and variable fixing mechanisms, and linearize the objective function to improve solution times of scenario subproblems. Using instances based on data from a major oncology hospital, we compare the performance of PHA with heuristics used in the relevant studies from the appointment scheduling literature. Our experiments provide insight into the issues related to the following questions:

1. What is the value of considering uncertainty in pre-medication and infusion durations when scheduling chemotherapy appointments?

2. What is the potential benefit of using the PHA over commonly used heuristics from the relevant appointment scheduling literature?

3. Which PHA routines and parameters must be more carefully designed to enhance the algorithm performance and solution quality?

4. How does the number of nurses and chairs affect optimal schedules in terms of patient waiting time, chair idle time and nurse overtime?

The organization of the remaining sections is as follows. In the next section, literature review and the main contributions of the article are presented. In Section 3, the problem description and model formulation are given. In Section 4, the proposed progressive hedging algorithm and its implementation details are presented. Our computational experiments and their results are demonstrated in detail in Section 5. Finally, concluding remarks and further research directions are discussed in Section 6.

2 Literature Review

Our literature review mainly focuses on outpatient chemotherapy scheduling problems. The reader is referred to Lame et al. (2016) for an extensive review on the subject. A detailed review of outpatient scheduling studies on primary care and specialty care can be found in Gupta and Denton (2008), Cayirli and Veral (2003), Ahmadi-Javid et al. (2017). However, before we proceed with the review of chemotherapy scheduling literature, we discuss studies on general outpatient appointment scheduling.

2.1 General Outpatient Scheduling Problems

Among the studies on outpatient scheduling, the ones that consider (i) patient sequencing (ii) appointment time decisions, or (iii) patient scheduling in a multiple resource setting are relevant to our study. As also pointed out in Ahmadi-Javid et al. (2017), most of the articles assume a predetermined patient sequence due to the complexity of sequencing problem (Kong et al., 2019; Begen and Queyranne, 2011; Denton and Gupta, 2003). In the context of single-server scheduling,

only a few articles consider sequencing and time setting decisions simultaneously (Berg et al., 2014; Mak et al., 2014; Mancilla and Storer, 2012; Denton et al., 2007). Among these studies, Berg et al. (2014) , Mancilla and Storer (2012), Denton et al. (2007) formulated stochastic programming models. Note that the second-stage subproblems in those models are much simpler than those of the chemotherapy scheduling problem due to the consideration of two types of resources in the latter.

In the literature of outpatient scheduling, the articles that consider multiple resources in the model are the most relevant ones to our study (Batun et al., 2011; Perez et al., 2013, Riise et al., 2016, Alvarez-Oh et al., 2018, Klassen and Yoogalingam, 2019; Leeftink et al., 2019; Hur et al., 2020). Below, we provide an in-depth comparison between our study and the studies that formulate two-stage stochastic programming model.

Batun et al. (2011) considered both surgeons and ORs in their study. However, the only assignment decision they made is the assignment of surgeries to ORs in their two-stage SP model. In other words, they assumed that the surgery list of a surgeon is known. Furthermore, the surgery-to-OR assignment decision was made at the first stage of their SP model. This helped them model the second-stage problem using only continuous variables. Due to the resulting structure of the model, they were able to implement the L-shaped algorithm (Van Slyke and Wets, 1969) to solve their model. On the other hand, in our model we make assignments to two different types of resources including nurse and chairs. Furthermore, the assignment decisions are made at the second stage of the SP formulation. These variables make the structure of the second-stage problem of our model different from that of Batun et al. (2011). The types of variables at each stage of an SP formulation are critical in determining the candidate decomposition algorithms to be implemented. Due to the binary and continuous variables in the second stage, the L-shaped algorithm is not a viable solution method for our model. Hence, we implement a progressive hedging algorithm to obtain solutions.

Perez et al. (2013) considered multiple equipment types and human resources while scheduling nuclear medicine appointments. They formulated both deterministic and stochastic formulations for the scheduling problem. They consider uncertainty in future patient arrival requests in their SP model but ignore any uncertainty in procedure durations. Note that the scheduling decisions in our SMIP formulation are made under uncertainty in treatment durations. Their SP formulation was designed to provide an appointment time for only one patient. To schedule multiple patients, they needed to solve the formulation multiple times within an algorithm. Their objective function

includes the waiting time from the arrival time of the appointment request until the planned time of appointment. However, our model considers the waiting time from the time of patient arrival to clinic on the day of appointment until the actual treatment start time on the same day. Furthermore, we consider resource overtime as well. Perez et al. (2013) assigned patients to small time slots which allowed them to have only binary variables in both stages of the formulation. However, we do not use time slots in our formulation, and instead model appointment time decisions using continuous variables, resulting in a formulation with mixed binary and continuous variables in both stages. Having only binary variables in the first-stage problem would allow us to implement an integer L-shaped algorithm (Laporte and Louveaux, 1993) or an evaluate-and-cut method proposed by Ahmed (2013) to obtain optimal solutions. However, due to the aforementioned structure of our model, we implement a modified PHA. We do not use time slots, since modeling appointments using time slots would lead to unnecessary idle times.

Alvarez-Oh et al. (2018) studied the allocation of patients to resources in a system with multiple service steps in the context of primary care clinic. However, they considered only patient-to-nurse assignments as the physician responsible for a patient is given in the problem. They also assigned patients to slots to determine appointment times, which allowed them to have only binary variables in the first stage of their SP formulation. Assignment of patients to multiple resources and the way appointment times are set constitute the main differences between our study and Alvarez-Oh et al. (2018).

Hur et al. (2020) scheduled patients of a multidisciplinary outpatient clinic using SP formulation. They assigned patients to multiple providers to be seen during their visit at a back pain integrated practice unit. However, they assumed that each patient is seen by only one provider at a time. The appointment times were also set in terms of time blocks, which results in an SP formulation having only binary variables at the first stage. Note that in our case the patients are assigned to multiple resources simultaneously in the pre-medication step. Continuous appointment times make our SP model structure different from that of Hur et al. (2020) as well.

Leeftink et al. (2019) also considered a multidisciplinary outpatient clinic to schedule appointments. However, their SP model is quite different from ours because they did not make individual patient assignments to resources. They instead created a blueprint schedule by determining the number of patients seen by each provider at each time period.

2.2 Chemotherapy Scheduling Problems

We do not restrict our review to studies that focus only on the treatment phase of chemotherapy (i.e. pre-medication and infusion). Some studies also consider the steps that need to be completed before the treatment starts, namely blood tests, oncologist evaluation, and drug preparation in the pharmacy. A patient's blood test result determines whether or not the patient is ready to go through the treatment. Once the blood test results are received from the laboratory, the treatment of the patient is either approved or postponed after the oncologist evaluates the results. In case the treatment is approved, a drug preparation order is provided to the pharmacist. The treatment can start when the chemotherapy drug is prepared.

Streams of literature most relevant to the work in this paper include the deterministic and stochastic versions of chemotherapy scheduling problems and their solution approaches, which are discussed in the remainder of this section.

2.2.1 Deterministic Chemotherapy Scheduling Problems

Turkcan et al. (2012) handled chemotherapy planning and scheduling problems in a hierarchical manner. In their first formulation, the aim is to minimize treatment delays while assigning the new patients to treatment days. After determining the daily patient lists, the authors focused on the daily patient scheduling problem by considering resource availabilities and acuity levels of the patients. Santibanez et al. (2012) suggested that the planner should consider the preferences of patients, capacity of the pharmacy and laboratory, schedules of the oncologists, and overtime for nurses while assigning appointment times to patients. To determine the chemotherapy schedules, a timetable was prepared in Dobish et al. (2003), where it was assumed that laboratory tests and oncologist evaluation are completed on the previous day of the treatment. Pharmacist availability was an important concern in the scheduling process. Oncologist evaluation and pharmacy preparation stages were studied in addition to the treatment stage in the model developed in Sadki et al. (2011). Nurses were always assumed to be ready for the treatment, so the scarce resources were the oncologists and chemotherapy chairs.

Heshmat et al. (2018) proposed a two-stage solution approach for the chemotherapy appointment scheduling problem. In the first stage, patients were grouped using clustering algorithms. In the latter stage, these clusters of patients were assigned to nurses, chairs, and time slots by solving a mathematical programming model. Huggins et al. (2014) developed a mathematical programming

model to maximize chair utilization while considering the workloads of the pharmacists and nurses. They validated the model solutions with a simulation study. Two heuristics were developed to assign patients to the infusion chairs by Sevinc at al. (2013). They decomposed the problem into two phases. They first determined the daily patient list for laboratory tests. The patients who obtain an oncologist approval according to their laboratory test results are then assigned to infusion chairs in the second phase. The makespan and weighted flow time were minimized in Heseraki et al. (2018) to schedule appointments of the chemotherapy patients. The authors expressed treatment durations in terms of time slots by assuming that a nurse can proctor the pre-medication of only one patient at a given time. Other aspects considered in the study are the patient priorities and lunch-coffee breaks of the nurses.

Liang and Turkcan (2016) discussed two different care delivery models for treatment of patients, namely *functional* and *primary care delivery models*. In the functional care delivery model, it is allowed to assign a different nurse to a patient at each treatment visit. On the other hand, a specific nurse is responsible for the patient's treatment in the primary care delivery model. The workload among nurses tends to show higher variability in the primary care delivery model, compared to the functional care delivery model. The authors constructed a mathematical programming model to minimize nurse overtime and total excess workload for the case of primary care delivery model. The acuity levels of patients are taken into account, and there is an upper bound on the assigned acuity level for each nurse in a time slot. Skill levels of the nurses are also included in the model. For the functional care delivery model, they aimed to minimize the total nurse overtime and patient waiting time within a multi-objective optimization model framework.

Our study differs from the studies summarized in this section since we consider uncertainty in the pre-medication and infusion durations.

2.2.2 Stochastic Chemotherapy Scheduling Problems

Alvarado and Ntaimo (2018) formulated three different mean-risk stochastic integer programming models for the chemotherapy appointment planning and scheduling problem. Acuity levels, treatment durations, and nurse availability were represented by stochastic parameters in their problem description. Their model was formulated to provide a solution for a single patient. In other words, the authors assumed that the appointments of earlier patients were already booked. They made their planning and scheduling decision for the current patient by considering the remaining time slots in the future days. The objective was to minimize the deviation from the target treatment

start day for the new patient, patient waiting time, and nurse overtime. In our study, we consider planning the daily schedules of multiple patients without using time slots.

Hahn-Goldberg et al. (2014) considered the uncertainty due to the arrivals of appointment requests and last minute changes in a schedule. They minimized makespan in the dynamic scheduling problem using an online algorithm. However, they did not consider uncertainty in treatment durations. A discrete event simulation model was developed to observe the effect of different operational decisions on the patient waiting time, clinic overtime, and resource utilization in an outpatient oncology clinic in Liang et al. (2015). The stochastic elements in the simulation model included unpunctual arrivals of patients and all service durations. However, the uncertainty was not considered in the mathematical programming model used to create schedules.

Tanaka et al. (2011) assigned patients to the infusion chairs using online bin packing heuristics. It was assumed that each infusion chair has a separate patient list for the treatments. The uncertainty in drug preparation and nursing durations before the infusion, such as taking vitals and assessment was taken into account. However, the study assumed deterministic pre-medication and infusion durations. Mandelbaum et al. (2016) considered uncertain treatment durations and unpunctual patients. They constructed a data-driven approach for the problem of patient appointment scheduling, assuming that the infusion chairs are the only servers of the system. Therefore, the impact of limited nurse availability on a schedule is not considered in this study.

The study most similar to that in this paper was conducted by Castaing et al. (2016), where the authors constructed a two-stage stochastic integer program to determine patient appointment times in an outpatient chemotherapy clinic in order to minimize the expected patient waiting time and total time required for all treatments (i.e., clinic closure time). The authors assumed that an initial schedule was already created. Their model allowed to make small revisions on the existing schedule as they assumed the sequence of patients cannot be changed. They set appointment times for a given sequence of patients and considered multiple chairs, but assumed a single nurse in the clinic. The binary variables in the second stage made the model difficult to solve. Therefore, a two-phase heuristic which facilitates setting the values of those variables was implemented to solve the problem. Our work differs from Castaing et al. (2016) in a number of ways. First, we do not make the restricting assumption that the patient sequence is fixed. Hence, our model does more than the refinement of existing schedules; it can be used to create an initial schedule. Moreover, we consider that there are multiple nurses in the clinic and that a patient can be assigned to any of the available nurses for treatment. Therefore, our model is applicable to clinics operating based

on a functional care delivery model. On the other hand, the model of Castaing et al. (2016) can be used by clinics functioning according to primary care delivery model. Furthermore, their model must be solved separately for each nurse, as it considers a single nurse.

3 The Chemotherapy Appointment Scheduling Problem

In this section, we define the Chemotherapy Appointment Scheduling Problem based on our observations of the chemotherapy operations at the Hacettepe University Oncology Hospital in Ankara, Turkey, which is one of the prominent oncology centers in the country. We also describe the proposed two-stage stochastic mixed integer programming formulation in detail.

3.1 Chemotherapy Operations in the Hacettepe Outpatient Chemotherapy Unit

The Hacettepe Outpatient Chemotherapy Unit consists of 32 chemotherapy chairs, 28 of which are used for treatments. The remaining four chairs are reserved for supportive care. The number of patients receiving treatment varies between 60 and 85 on a given day. On average, 10 nurses, including a head nurse, provide service to patients each day. During chemotherapy planning (which is outside the scope of this paper), patients are assigned to each day based on the day of their previous visit, estimated duration of their treatment, and the expected treatment duration of patients assigned to each upcoming day. A daily shift starts at 8:00 and ends at 17:00 (unless overtime is needed), with a lunch break between 12:00-13:00. The daily schedule of patients is arranged by the head nurse, who assigns appointment times, chairs and nurses to the patients.

Our observations at the chemotherapy unit aimed to capture the operations conducted until and during the chemotherapy treatment. The blood tests and oncologist evaluations are carried out on a different day before the treatment. The results of these tests are examined by the oncologist to decide whether the patient has convenient health status to receive a chemotherapy treatment. When the oncologist approves the treatment protocol, she updates or confirms the dosages of the drugs. The drugs are then prepared in the pharmacy lab before the patient arrives at the hospital for her treatment. During each treatment visit, a patient receives two types of medications: (i) pre-medication drugs, which are injected prior to the chemotherapy infusion to help prevent side effects; and (ii) chemotherapy infusion drugs, which are used for cancer treatment.

There are certain time slots that the patients are assigned to according to their estimated treatment duration, which is predicted by the chemotherapy unit according to the types and dosages

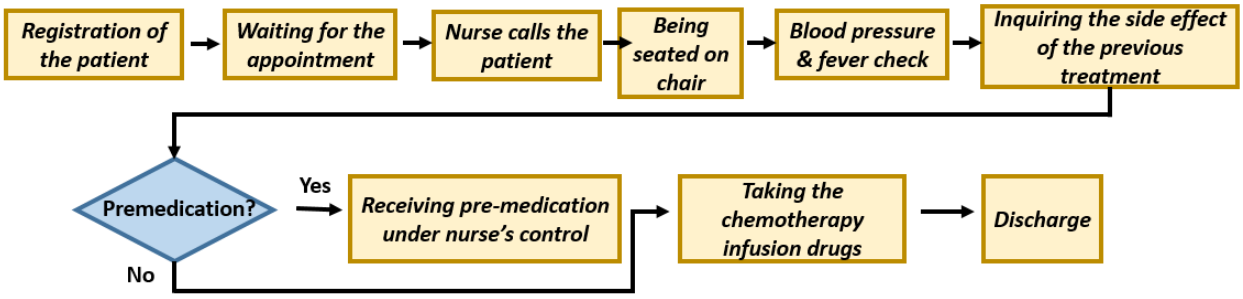

Figure 1: Patient flow chart for treatment process

of the drugs injected. During a day, the unit makes use of four time slots: 8:00-10:30, 10:30-12:00, 13:00-15:30 and 15:30-17:00. If the estimated treatment duration of a patient is less than the length of a slot, the patient is given an appointment time so that the expected end of treatment is within the same slot. Otherwise, the patient can be assigned to more than one slot. If the estimated treatment duration of a patient is longer than those of other patients, the head nurse prefers to assign this patient to the beginning of the workday in order to prevent excessive overtime. In the rare case that the estimated duration of the treatment exceeds 270 minutes, the appointment is generally split into two sessions, one in the morning and one in the afternoon. Given the uncertainties in treatment times, the three main objectives in designing the schedule are to ensure that (i) each patient can actually start their treatment after experiencing minimal or no wait, (ii) all chairs are highly utilized, and (iii) each nurse spends minimal or no time above regular shift duration.

The treatment of a patient consists of a sequence of events, as illustrated in Figure 1. A patient that arrives at the unit is immediately registered and becomes available for treatment at the appointment time. If a nurse and/or chair is not available at the time, the patient has to wait for the treatment. A nurse calls the patient when an infusion chair and the nurse both become available. After the patient is seated in a chemotherapy chair, the nurse measures the fever and blood pressure, and establishes a vascular access to start the pre-medication process. Aside from rare cases where the patient may not need pre-medication drugs, the infusion process starts after the pre-medication process ends. When the infusion process is over, the patient is discharged, and the nurse calls the next patient to a chair that has become available. In general, each nurse may be responsible for up to four patients at a given time. The Hacettepe Outpatient Chemotherapy Unit applies a modified form of the *functional care delivery*, where in line with this procedure each patient can be assigned to different nurses in subsequent visits. However, nurse-patient assignments in the clinic also consider the type of cancer and expected duration of treatment, which may limit

the pool of nurses that can be assigned to the patient. The balance of both criteria is aimed to be satisfied to create equity between nurses.

3.2 A Two-Stage Stochastic Mixed-Integer Programming Model for the Chemotherapy Appointment Scheduling Problem

We formulate a two-stage stochastic mixed-integer programming (SMIP) model for the Chemotherapy Appointment Scheduling Problem, motivated by the operations at the Hacettepe Outpatient Chemotherapy Unit. In particular, we study the problem of sequencing patients and setting appointment times for a chemotherapy unit by considering the availability of nurses and chairs under uncertainty on the pre-medication and infusion durations. We sequence patients of a daily appointment list, set appointment times, and assign patients to nurses and chairs in our model. We consider an objective function that minimizes a weighted combination of patient waiting time, chair idle time and nurse overtime.

In the remainder of this section, we provide our assumptions and two-stage stochastic programming model and propose a number of valid inequalities and bounds to strengthen the model.

3.2.1 Two-Stage Stochastic Mixed-Integer Programming Formulation

In this section, we discuss our two-stage SMIP model for the chemotherapy appointment scheduling problem. The following assumptions are made in the model:

- We take the decisions in the chemotherapy planning phase (appointment of patients to days) as given and focus on the chemotherapy scheduling decisions for each day.
- Among the events shown in Figure 1, we focus on the three main events in between the patient appointment time and treatment completion time. These events include patient waiting, pre-medication, and infusion.
- We do not consider time slots (as applied in the Outpatient Chemotherapy Unit) while determining the patient appointment times. Although the use of slots facilitates the construction of schedules manually, having to fit the treatment of patients into pre-determined slots may be too restrictive, as it may lead to long idle times within some slots. To overcome this, our model schedules patients in any minute within each half-day shift.

- Patients can be treated by any available nurse, as is the case for *functional care delivery*. Nurses are also assumed to have identical skills.
- A nurse can perform only one patient's pre-medication process at any time.
- A nurse can proctor the infusion process of multiple patients while conducting the pre-medication process of a patient.
- Patients become ready exactly on the appointment time for treatment.
- The sequence of patients is not allowed to change after they arrive at the chemotherapy unit.

At the first stage of the model, patients are sequenced and their appointment times are set. These constitute the main decisions in the model. Then, pre-medication and infusion durations are realized. At the second stage, patients are assigned to nurses and chairs. The remaining decision variables at the second stage help determine patient waiting time, chair idle time, discharge time, and nurse overtime. Overtime is calculated separately for each nurse that works for more than the planned shift length. Chairs become idle due to two reasons. First, treatment of a patient may finish earlier than the appointment time of the subsequent patient. Second, longer than expected durations of pre-medications may prevent nurses from initiating the treatment of a waiting patient. Waiting occurs when the treatment starts later than the appointment time of the patient, which may happen when pre-medications or infusions for the previous patients may last longer than expected. The unavailability of nurses or chairs may also cause waiting.

We assume a finite set of scenarios representing uncertainty in pre-medication and infusion durations. Given these scenarios, we formulate the following two-stage SMIP model for our problem based on sets, parameters, first and second-stage decision variables summarized in Table 1.

INSERT TABLE 1 HERE

$$min \quad \mathcal{Q}(\mathbf{a},\mathbf{b}) \tag{1}$$

$$b_{ij} + b_{ji} = 1 \qquad \forall i, j \in I, j > i \tag{2}$$

$$a_j \geq a_i - M(1 - b_{ij}) \qquad \forall i, j \in I, j \neq i \tag{3}$$

$$b_{ij} \in \{0, 1\} \qquad \forall i, j \in I, j \neq i \tag{4}$$

$$a_i : integer \qquad \forall i \in I \tag{5}$$

where

$$\mathcal{Q}(\mathbf{a},\mathbf{b}) = E_{\xi}[\mathrm{Q}(\mathbf{a},\mathbf{b},\xi(\omega))]$$

is the expected recourse function, and

$$\mathrm{Q}(\mathbf{a},\mathbf{b},\xi(\omega)) = \min\left\{\lambda_1 \sum_{i\in I} w_i^{\omega} + \lambda_2 \sum_{n\in N} O_n^{\omega} + \lambda_3 \sum_{c\in C} I_c^{\omega}\right\}$$

$$\sum_{n\in N} x_{in}^{\omega} = 1 \qquad \forall i \in I \quad (6)$$

$$\sum_{c\in C} y_{ic}^{\omega} = 1 \qquad \forall i \in I \quad (7)$$

$$a_i + w_i^{\omega} + s_i^{\omega} + t_i^{\omega} = d_i^{\omega} \qquad \forall i \in I \quad (8)$$

$$a_j + w_j^{\omega} \geq a_i + w_i^{\omega} + s_i^{\omega} - M(3 - b_{ij} - x_{in}^{\omega} - x_{jn}^{\omega}) \qquad \forall i,j \in I, j \neq i, \forall n \in N \quad (9)$$

$$a_j + w_j^{\omega} \geq d_i^{w} - M(3 - b_{ij} - y_{ic}^{\omega} - y_{jc}^{\omega}) \qquad \forall i,j \in I, j \neq i, \forall c \in C \quad (10)$$

$$a_j + w_j^{\omega} \geq a_i + w_i^{\omega} - M(1 - b_{ij}) \qquad \forall i,j \in I, j \neq i \quad (11)$$

$$O_n^{\omega} \geq d_i^{\omega} - H - M(1 - x_{in}^{\omega}) \qquad \forall i \in I, \forall n \in N \quad (12)$$

$$O_n^{\omega} \leq L \qquad \forall n \in N \quad (13)$$

$$T_c^{\omega} \geq H \qquad \forall c \in C \quad (14)$$

$$T_c^{\omega} \geq d_i^{\omega} - M(1 - y_{ic}^{\omega}) \qquad \forall i \in I, \forall c \in C \quad (15)$$

$$I_c^{\omega} \geq T_c^{\omega} - \sum_{i\in I}(s_i^{\omega} + t_i^{\omega}) y_{ic}^{\omega} \qquad \forall c \in C \quad (16)$$

$$x_{in}^{\omega} \in \{0,1\} \qquad \forall i \in I, \forall n \in N \quad (17)$$

$$y_{ic}^{\omega} \in \{0,1\} \qquad \forall i \in I, \forall c \in C \quad (18)$$

$$d_i^{\omega}, w_i^{\omega} \geq 0 \qquad \forall i \in I \quad (19)$$

$$O_n^{\omega} \geq 0 \qquad \forall n \in N \quad (20)$$

$$T_c^{\omega}, I_c^{\omega} \geq 0 \qquad \forall c \in C \quad (21)$$

The objective function in (1) only includes the expected second-stage function, since there is no contribution from the first stage to the objective function. The expected second-stage function aims to minimize the expected weighted sum of total patient waiting time, nurse overtime, and chair idle time. First-stage constraints are given by (2)-(5). Patient precedence relations are determined using constraints (2). If patient $i \in I$ is scheduled before the patient $j \in I$, the corresponding b_{ij} value should be equal to 1. Constraints (3) establish the relationship between the binary precedence

variables and appointment times, in that if patient $i \in I$ precedes patient $j \in I$ in the list, then the appointment time of patient $i \in I$ should be no later than that of patient $j \in I$. Binary and integrality restrictions on the first-stage variables are represented in constraints (4) and (5), respectively.

For each scenario representing a set of chemotherapy durations, a subproblem is solved to obtain second-stage decisions. The constraints for the second-stage scenario subproblems are expressed by (6)-(21). Constraints (6) and (7) enforce every patient to be assigned to exactly one nurse and one chair, respectively. Constraints (8) calculate the patient discharge time, which is equal to the sum of the patient appointment time, waiting time, and durations of pre-medication and infusion. Constraints (9) ensure that if patient $i \in I$ and $j \in I$ are assigned to the same nurse, and patient $i \in I$ is scheduled before $j \in I$, then the treatment start time of patient $j \in I$ should start after the end of pre-medication of patient $i \in I$. Similar relation also holds for chair-precedence relationship, which is demonstrated in constraints (10). If two patients are assigned to the same chair, the treatment of the latter may begin only after the discharge time of the former one. By constraints (11), a patient's treatment start time should be earlier than those of the patients that are scheduled later. Overtime of a nurse should be either zero or the difference between the discharge time of the last discharged patient assigned to this nurse and the shift length, as determined by constraints (12). In oncology units, it is desired to have limited overtime to enhance nurses' working conditions and preserve equity among nurses. A predetermined bound is included to ensure this in the model, which also tightens the feasible region of the formulation. The associated constraint is provided in equation (13). Constraints (14)-(16) are used to calculate idle time of the chairs in the clinic. The idle time of each chair is calculated considering the shift length and discharge time of the lastly scheduled patient who is treated in the associated chair. The maximum of the discharge time of the last patient and shift length is used to measure the idle time of the associated chair. The remaining constraints are the sign and binary restrictions on the second-stage decision variables.

Given that nurses and chairs are identical, assigning patients to them in the first stage may lead to inefficient schedules. In practice, assignment of each patient to chairs and nurses is generally made dynamically after arrival, well before the uncertainty on the pre-medication and infusion durations is fully realized. Although treating these second-stage decisions in our model may seem in contradiction to practice, we show that once the patient sequencing is fixed in the first stage, these decisions become straightforward. More formally, we use the following proposition, the proof of which is given in Appendix A.

Proposition 1 *Given the sequencing of patients from the first stage, assigning arriving patients to the first available chair and nurse produces the optimal schedule for the second stage of the Chemotherapy Appointment Scheduling Problem.*

Using this proposition, we are able to consider chair and nurse assignment decisions in the second stage without violating the dynamic nature of the optimal assignment policy. Note that a particular assumption, which is also represented by constraints (11), has a critical role in the proof of the proposition. Constraints (11) enforce that the appointment sequence in the first stage is the same as the treatment sequence in the second stage. This ensures that the optimal policy to assign patients to chairs and nurses is non-idling. Hence, there is no need to make such assignments dynamically when patients arrive.

The rule dictated by the above discussed assumption is valid from a practical point of view. As all patients have the same urgency levels and are assumed to arrive on time, the managers would not ask an arriving patient to wait and let his/her successor be treated first when a nurse and a chair are available in the unit. This approach would dramatically reduce the satisfaction of the waiting patient. Therefore, it may be impractical to reshuffle patient sequence after the patient arrivals.

We note here that although it is desired, we do not model a limit on the number of patients to be observed by a nurse at a given time explicitly. This potential issue is implicitly addressed by the model with the inclusion of the first two components of the objective function, as assigning too many patients to a single nurse at a given time may excessively increase the patient waiting time and nurse overtime. Hence, these two objective components would drive the solution to a balanced assignment of patients to nurses. In case an explicit limit B on the number of patients assigned to a nurse at a time is desired, the following constraints can be added to the two-stage SMIP model:

$$u^{\omega}_{ink} \geq x^{\omega}_{in} + \beta^{\omega}_{ink} + \gamma^{\omega}_{ink} - 2 \qquad \forall i \in I,\, n \in N,\, k \in K \tag{22}$$

$$M\beta^{\omega}_{ink} \geq k - a_i - w^{\omega}_i \qquad \forall i \in I,\, n \in N,\, k \in K \tag{23}$$

$$M\gamma^{\omega}_{ink} \geq a_i + w^{\omega}_i + s^{\omega}_i + t^{\omega}_i - k \qquad \forall i \in I,\, n \in N,\, k \in K \tag{24}$$

$$\sum_{i \in I} u^{\omega}_{ink} \leq B \qquad \forall n \in N,\, k \in K \tag{25}$$

$$u^{\omega}_{ink},\, \beta^{\omega}_{ink},\, \gamma^{\omega}_{ink} \in \{0,1\} \qquad \forall i \in I,\, n \in N,\, k \in K, \tag{26}$$

where K represents the set of discrete time units (e.g., minutes), β^{ω}_{ink} takes a value of 1 if patient

$i \in I$ has started treatment with nurse $n \in N$ before $k \in K$, γ^{ω}_{ink} is 1 if patient $i \in I$ has not finished treatment with the same nurse before $k \in K$, u^{ω}_{ink} represents whether patient $i \in I$ is being treated by nurse $n \in N$ at time $k \in K$, and M is a large number. It should be noted here that such a modeling approach would require discretization of the time periods (e.g., in minutes), and would substantially add to the complexity of the model. Furthermore, Proposition 1 should be modified to assign arriving patient to the first available nurse observing the treatment of less than B patients.

3.2.2 Improvements in the Formulation

Based on the preliminary runs, it is observed that solving even deterministic instances of the problem necessitates significantly long solution times. Consequently, symmetry-breaking constraints and valid bounds are added in order to strengthen the formulation and decrease the computational burden.

The first set of constraints relates to breaking symmetry with respect to the overtime of nurses. Since nurses are assumed to be identical, the schedules of the nurses are interchangeable. By convention, we enforce the overtime of nurse with index 1 to be no less than those of the other nurses. In this way, overtime may be assigned to nurses in non-increasing order of indices. The mathematical representation of this constraint is given by:

$$O^{\omega}_{n} \geq O^{\omega}_{n+1} \qquad \forall n \leq |N| - 1, \forall \omega \in \Omega \tag{27}$$

Note that balancing nurse workload is not among the main objectives in our model. However, the upper limit introduced by constraint (13) defines an acceptable bound and hence prevents excessive imbalance.

Next, a lower bound on nurse overtime is defined using the structure of the problem. If one can equally distribute the summation of treatment durations to chairs, the resulting average total treatment time per chair provides the smallest possible value for the maximum of patient discharge times. Therefore, the calculated average value is a lower bound for the sum of nurse shift time and overtime of nurse 1, since the overtime of nurse 1 is at least as high as all others according to equation (27). The mathematical representation of this constraint is shown given by:

$$O^{\omega}_{1} + H \geq \frac{\sum_{i \in I}(s^{\omega}_{i} + t^{\omega}_{i})}{|C|} \qquad \forall \omega \in \Omega \tag{28}$$

As proposed in Mancilla and Storer (2012), the appointment and waiting times of the first patient in the schedule can be assigned to zero to reduce the number of decision variables. According to the results of preliminary experiments, adding such time assignments does not have a significant effect to reduce the computation times, and thus these are not included in the formulation.

Equations (27)-(28) are added to the main model and used in the computational experiments.

4 Solution Methodology

SMIP models are in general computationally demanding mainly due to the large amount of variables and constraints depending on the number of scenarios (Birge, 2011). Decomposition based solution methods are generally utilized to solve such models in the literature. The well-known stage decomposition algorithm proposed by Slyke and Wets (1969), *L-shaped method*, cannot be applied on stochastic programs in case there are binary variables in the second stage as in our SMIP formulation. To handle those cases, Laporte and Louveaux (1993) extended the method and proposed the *integer L-shaped algorithm*, which can be implemented when only binary variables exist in the first stage. Other solution methods aim to overcome the difficulty associated with having binary variables in the second stage through approaches based on value function and set convexification (Sen, 2005; Yuan, 2009). These disjunctive decomposition-based branch-and-cut algorithms are known to be very effective, but they can also be applied on models with only binary variables in the first stage. On the other hand, in our formulation, the first stage includes both binary and general integer variables.

Scenario decomposition-based approaches are also frequently used to solve SMIP models. The Progressive Hedging Algorithm (PHA) has been increasingly preferred among them (Gul et al., 2015; Hvattum et al., 2009; Goncalves et al., 2012). The PHA, proposed by Rockafellar and Wets (1991), is based on an augmented Lagrangian relaxation technique. The algorithm decomposes the problem into single-scenario subproblems, and aims to obtain an *admissible* solution that is feasible for all scenarios by aggregating the scenario solutions at each iteration. It requires reformulation of the model to attain a structure appropriate for scenario decomposition. It then relaxes the constraints that enforce the first-stage decisions to be the same for every scenario in the reformulation and penalizes the violation of these constraints by introducing a Lagrangian term and quadratic penalty term in the objective function. The algorithm is shown to converge to the optimal solution for convex models. Since our model includes general integer and binary decision variables, it is

not convex. Therefore, PHA is used as a heuristic approach for the Chemotherapy Appointment Scheduling Problem. Note that there is significant evidence in the literature showing that the PHA is an effective heuristic for SMIP models (Crainic et al., 2011; Gul et al., 2015; Watson and Woodruff, 2011).

In the remainder of this section, general steps of the PHA are given, followed by a literature review on different applications of the PHA. We make use of a number of approaches from the literature to modify the PHA, which is discussed in the last part of this section.

4.1 The Progressive Hedging Algorithm

We first reformulate our SMIP model to facilitate scenario decomposition. In the reformulation for PHA, which we call as SMIP-R (see Appendix B for the model), *non-anticipativity constraints* are explicitly formulated. To formulate these constraints, a copy of a first-stage variable must be created for each scenario. The constraints then enforce the first-stage decision variables to be equal to each other across all scenarios to prevent anticipation of the future. Since the appointment time values (a_i) and the precedence relationship variables (b_{ij}) are dependent on each other, there is no need to include non-anticipativity constraints for the latter. In SMIP-R, non-anticipativity constraints for the appointment time values are given by:

$$a_i^\omega = a_i \quad \forall i \in I, \forall \omega \in \Omega, \tag{29}$$

where the a_i values represent the *consensus variable* for appointment time of patient $i \in I$. Prior to the PHA implementation, these non-anticipativity constraints are relaxed, and their violation is penalized using two terms in the objective function. In the first term, the amount of violation for each patient is multiplied by Lagrangian multipliers denoted by $\mu_i^\omega \ \forall i \in I, \forall \omega \in \Omega$. The second term is a quadratic component that takes the squared sum of these violations, and multiplies it by $\frac{\rho_1}{2}$, where ρ denotes the penalty parameter. The revised objective function is given by:

$$\begin{aligned}&\lambda_1 \sum_{i\in I}\sum_{\omega\in\Omega} p^\omega w_i^\omega + \lambda_2 \sum_{n\in N}\sum_{\omega\in\Omega} p^\omega O_n^\omega + \lambda_3 \sum_{c\in C}\sum_{\omega\in\Omega} p^\omega I_c^\omega + \sum_{i\in I}\sum_{\omega\in\Omega} \mu_i^\omega (a_i^\omega - a_i) \\ &+ \frac{\rho_1}{2} \sum_{i\in I}\sum_{\omega\in\Omega} ||a_i^\omega - a_i||^2 \end{aligned} \tag{30}$$

To obtain a separable formulation, variables a_i in (30) are replaced by $\hat{a}_i$, a consensus parameter that is equal to the weighted average of appointment times over all scenarios, where the weights

are equal to the probabilities of scenarios.

$$\hat{a}_i = \sum_{\omega \in \Omega} p^\omega a_i^\omega \quad \forall i \in I \tag{31}$$

The resulting structure of the objective function and constraint set allow the SMIP-R to be decomposed into multiple scenario subproblems, which are solved independently at each iteration of the algorithm.

The general steps of the PHA, shown also in Algorithm 1, are as follows: In step 1, algorithm parameters are initialized. Penalty parameter of the quadratic term, denoted by ρ, is set as a positive constant value of ρ^0, and the Lagrangian multipliers $\mu_i^{\omega(v)}$ are initialized as 0 in the first iteration. In step 2, each scenario subproblem is solved. Note that quadratic penalty and Lagrangian terms are ignored in the first iteration of this step. In step 3, the appointment time values obtained for every patient and scenario are used to calculate the consensus parameters $\hat{a}_i$ for every patient. In step 4, the penalty parameter ρ is updated by multiplying it with a pre-determined positive constant α. By using the value of the penalty parameter, Lagrangian multipliers are updated considering the distance between scenario solutions and the consensus parameter value. At the end of each iteration, the termination criterion is checked. If all first-stage solutions are identical, the algorithm stops. Otherwise, the algorithm returns to step 2 and solves scenario subproblems with the updated values of the multipliers and consensus parameter.

4.2 Previous Studies on the Progressive Hedging Algorithm

Hvattum et al. (2009), Gul et al. (2015), Gade et al. (2016), Watson et al. (2011), Cheung et al. (2015), Mulvey and Vladimirou (1991), Goncalves et al. (2012), Atakan et al. (2018), and Helgason et al. (1991) used PHA in application areas such as inventory-routing, surgery scheduling, unit commitment, hydrothermal systems, and fisheries management. A number of modifications to improve convergence behavior and computation times of the PHA were suggested by Watson and Woodruff (2011), who classified the changes made in the algorithm in four categories: computing effective ρ values, accelerating convergence, termination criteria and detecting cyclic behaviour. Cheung et al. (2015) solved scenario subproblems approximately in the early iterations of the PHA, since obtaining exact solutions from the scenario subproblems is not a necessity for the algorithm. PHA was extended with some innovations such as dynamic multiple penalty parameters and heuristic intermediate solutions in Hvattum et al. (2009), where the penalty parameter is increased

when the distance between the scenario solutions and the consensus parameter increases. When the consensus parameters at consecutive iterations moves farther away from each other, the penalty parameter is reduced. Mulvey and Vladimirou (1991) also applied a dynamic penalty parameter update method. The technique can prohibit stalling and ill-conditioning solution structures, which occurs when the algorithm gets stuck at suboptimal solutions.

Algorithm 1 Progressive Hedging Algorithm

Step 1: Initialization:
Let v=1
$\rho_1{}^{(v)} = \rho_1{}^0$
$\mu_i^{\omega(v)} = 0 \quad \forall i \in I, \forall \omega \in \Omega$
Step 2: Solving scenario-subproblems:
if $v = 1$;
For each $\omega \in \Omega$; ignore Lagrangian and quadratic penalty terms and solve
scenario subproblems to obtain:
$a_i^{\omega(v)} \quad \forall i \in I, \forall \omega \in \Omega$
end if
else
For each $\omega \in \Omega$; solve scenario subproblems to obtain:
$a_i^{\omega(v)} \quad \forall i \in I, \forall \omega \in \Omega$
end else
Step 3: Calculate the consensus parameters:
$\hat{a_i} = \sum_{\omega \in \Omega} p^\omega a_i^{\omega(v)} \quad \forall i \in I$
Step 4: Update penalty parameters:
if $v > 1$;
$\rho^{(v+1)} = \alpha \rho^{(v)}$
end if
Step 5: Update Lagrange Multipliers:
$\mu_i^{\omega(v+1)} = \mu_i^{\omega(v)} + \rho^{(v)}(a_i^{\omega(v)} - \hat{a_i}) \quad \forall i \in I, \forall \omega \in \Omega$
Step 6: Termination:
if all first stage solutions are the same, then stop.
$a_i^{\omega(v)} = \hat{a_i} \quad \forall i \in I, \forall \omega \in \Omega$
end if
else
Set v= v+1;
end else
return to Step 2

4.3 Modifications on the Progressive Hedging Algorithm

When the PHA is applied to the Chemotherapy Appointment Scheduling Problem, even single-scenario subproblems require substantial computational times, which deem realistically-sized in-

stances of the problem impossible to solve in reasonable time. A number of issues contribute to this issue. The first one is the quadratic term used in the objective function of the subproblems, whereas the second one is the choice of penalty parameter update method. We investigate other improvement ideas on the PHA by detecting and favoring solutions found in the majority of the scenario subproblems, detecting cycles, and varying the optimality gap parameter of CPLEX to obtain approximate scenario subproblem solutions to decrease solution times. We also propose a lower bounding scheme for nurse overtime by benefiting from the structure of the PHA. In the following sections, these modifications on the PHA are presented.

4.3.1 Handling the Quadratic Term in the Objective Function

The scenario subproblems solved at each iteration of the PHA includes a quadratic term in the objective function, which makes them difficult to solve. This term can be expanded as:

$$\frac{\rho}{2}\sum_{i\in I}\sum_{w\in\Omega}||a_i^\omega-\hat{a}_i||^2=\frac{\rho}{2}\sum_{i\in I}\sum_{\omega\in\Omega}(a_i^\omega)^2-\rho\sum_{i\in I}\sum_{\omega\in\Omega}(a_i^\omega\hat{a}_i)+|\Omega|\frac{\rho}{2}\sum_{i\in I}\hat{a}_i{}^2 \tag{32}$$

Since $\hat{a}_i$ is a parameter, the only quadratic part is the first term on the right-hand side of (32). Our aim is to use a method to handle this term in a fast manner without sacrificing significantly from the solution quality. For this end, we experiment with three different approaches: (1) quadratic programming using commercial solvers, (2) second-order cone programming (SOCP), and (3) linearization of the quadratic term.

For the first approach, the quadratic term is included in the objective function without making any change in the algorithm. In the second approach, additional variables are defined to convert the quadratic term into an appropriate format for second-order cone programming. For the third approach, two alternatives are available in the PHA literature. The quadratic term is linearized using a piecewise linear function in Watson et al. (2012). Alternatively, Helseth (2016) dynamically approximated the quadratic term by benefiting from tangent line approximation. In particular, linear cuts that approximate the quadratic term are added to scenario subproblems to obtain a linear objective function.

Based on our preliminary experiments, we use the third approach, since it outperforms the other two in terms of the computational times significantly, without substantial sacrifice in terms of solution quality. Due to the use of linearized penalty components in the objective function, we call our algorithm as the *linearized progressive hedging algorithm* (LPHA). The linearization

method that we use, which is based on the approach in Helseth (2016), is explained next.

A general form of Taylor's approximation, $f(m) \approx f(n) + f'(n)(m-n)$, can be used to derive a cut that provides a lower bound for the nonlinear term. If the cut is generated at different operating points (n) iteratively, a close approximation of $(a_i^\omega)^2$ is obtained after some iterations.

In our case, at iteration v, the value of $(a_i^\omega)^2$ is approximated based on the operating point $(a_i^{\omega(v-1)})$ which represents the appointment time set for patient i in scenario ω at the previous iteration. Note that an alternative operating point would be selected as $\hat{a}_i$ calculated in iteration $(v-1)$. However, we choose the former alternative based on the results of preliminary experiments. After replacing $(a_i^\omega)^2$ by a new variable g_i^ω, the proposed linear cuts are then formulated as follows:

$$(g_i^\omega) \geq (a_i^{\omega(v-1)})^2 + 2(a_i^{\omega(v-1)})(a_i^\omega - a_i^{\omega(v-1)}) \tag{33}$$

The cuts in (33) are added to scenario subproblems at each iteration until it is noticed that the value of g_i^ω does not change from one iteration to another.

4.3.2 Choice of the Penalty Parameter Update Method

In the PHA, the penalty parameter ρ is updated by multiplying it with constant α in every consecutive iteration. When the multiplier α is set to 1, the penalty parameter becomes stable in the algorithm. Previous studies have shown that low values of ρ tend to improve the quality of the solutions at the expense of longer computational times. On the other hand, high values of the penalty parameter can lead to oscillations in the solutions and the algorithm generally converges faster to a low-quality solution. Therefore, it is crucial to find a balance between solution time and quality using an appropriate level of the penalty parameter over iterations. Designing a well-performing penalty update method is challenging due to this trade-off. Besides, there is no straightforward way to select the initial value of the penalty parameter; it is mostly dependent on the objective function coefficients and scales of the decision variable values in the problem.

One way to resolve the trade-off between solution quality and computational time is to apply a dynamic penalty parameter scheme by updating the value of the parameter in subsequent iterations. In this study, we use a method inspired by the approach discussed in Hvattum and Lokketangen (2009). In this method, two parameters are made use of by exploiting the relationship between the primal and dual. The first parameter, denoted by Δ_p measures the square of the difference between consensus parameters for the appointment time of each patient at consecutive iterations.

If Δ_p is increasing, this means that consensus parameter is changing at a greater rate. This also implies that the current value of consensus parameter is not promising, and therefore it is risky to enforce convergence immediately. The second parameter, denoted by Δ_d, compares the convergence behaviour of the first-stage variable to the consensus parameter at consecutive iterations. If Δ_d is increasing, this indicates that the first-stage variables in different scenario subproblems are farther away from the consensus parameter.

In our implementation of the penalty parameter update method, when Δ_d increases in consecutive iterations, penalty parameter ρ is multiplied by a constant α that is greater than 1. By this way, the algorithm is more inclined to converge faster. If the difference between Δ_d in consecutive iterations is not positive, the behaviour of Δ_p is observed. If Δ_p is increasing, the penalty parameter is reduced by a factor of $\frac{1}{\alpha}$. If both consecutive Δ_p and Δ_d differences are negative, the current penalty parameter is preserved.

Based on preliminary results, we observe that penalty parameter can quickly climb to very large values as a result of multiplying with α. In terms of convergence, higher penalty parameter values lead the algorithm to prematurely converge to suboptimal solutions. To avoid this issue, the value of penalty parameter is bounded above by a pre-determined parameter ρ^u, so that the algorithm can converge to a higher quality solution.

It is also observed in our preliminary experiments that keeping the penalty parameter small is beneficial in terms of solution quality. However, if the predefined bound for the penalty parameter is kept smaller than needed during the iterations, this may result in excessive computational times. To avoid this, we preserve a small bound at the earlier iterations to obtain high quality solutions, and then we increase the limit in the later iterations to achieve fast convergence. The details of the penalty update method is provided in Algorithm 2. In the algorithm, ρ_1^u and ρ_2^u refer to two different penalty parameter limits, where $\rho_1^u < \rho_2^u$. Furthermore, *iterlimit* corresponds to the iteration value at which the upper bound is changed from ρ_1^u to ρ_2^u.

4.3.3 Cycling and Majority Value Detection

Cycles are frequently observed within a typical implementation of the PHA. This behaviour may prevent convergence. We benefit from the idea of cycling prevention suggested by Watson and Woodruff (2011) to overcome this. The appointment time values may appear identical over several consecutive PHA iterations. The direct consequence of this is that the consensus parameter values remain constant over the iterations. On the other hand, Lagrangian multipliers change in every

Algorithm 2 Penalty Update Method with Limit

$\Delta_p{}^{(v)} = \sum_{i \in P} (\hat{a_i}^{(v)} - \hat{a_i}^{(v-1)})^2$
$\Delta_d{}^{(v)} = \sum_{i \in P} \sum_{\omega \in \Omega} (a_i^{\omega(v)} - \hat{a_i}^{(v)})^2$
if $v \leq iterlimit$
    $\rho^u = \rho_1^u$
else
    $\rho^u = \rho_2^u$
if $\Delta_d{}^{(v)} - \Delta_d{}^{(v-1)} > 0 \quad \& \quad \rho^{(v)} < \rho^u$;
    $\rho^{(v+1)} = \alpha \rho^{(v)}$
elseif $\Delta_d{}^{(v)} - \Delta_d{}^{(v-1)} > 0$
    $\rho^{(v+1)} = \rho^u$
elseif $\Delta_p{}^{(v)} - \Delta_p{}^{(v-1)} > 0$
    $\rho^{(v+1)} = \frac{1}{\alpha} \rho^{(v)}$
elseif $\rho^{(v)} \leq \rho^u$;
    $\rho^{(v+1)} = \rho^{(v)}$
else $\rho^{(v+1)} = \rho^u$

iteration, since we update their values using Step 5 of Algorithm 1.

While updating the Lagrangian multipliers, the differences between appointment time values and consensus parameter are penalized by multiplying it with a penalty parameter. Due to this reason, Lagrangian multipliers vary over iterations even if the appointment time and consensus parameter values stay the same. Hence, any cycling behaviour of the algorithm can be detected by examining the behaviour of the Lagrangian multipliers.

Watson and Woodruff (2011) used a hashing scheme to detect cycles. In our case, we would like to detect cycles in integer appointment times for each patient. Therefore, we have to check the behavior of the Lagrangian multipliers associated with each patient. In their approach, Watson and Woodruff (2011) first generated hash weights for each scenario (z_ω). In our algorithm, hash weights are the random numbers between 0 and 1. These hash weights are then multiplied with the Lagrangian multipliers (μ_i^ω). Hash value for each patient is calculated in every iteration as:

$$h_i = \sum_{\omega \in \Omega} z_\omega \mu_i^{\omega(v)} \quad \forall i \in I \tag{34}$$

Note that $h_i = 0$ at iteration $v = 1$, as $\mu_i^{\omega(1)} = 0$. If z_ω values are set equal to the probability of occurrence of each scenario (p^ω), then h_i would be zero for $v > 1$ due to the reason briefly explained next. Since $\mu_i^{\omega(v+1)} = \rho^{(v)}(a_i^{\omega(v)} - \sum_{\omega \in \Omega} p^\omega a_i^{\omega(v)})$ for all $i \in I, \omega \in \Omega$ at $v = 1$, $\sum_{\omega \in \Omega} p^\omega \mu_i^{\omega(2)} = 0$.

By induction, it can also be verified that $\sum_{\omega \in \Omega} p^{\omega} \mu_i^{\omega(v)} = 0$ for all v. Therefore, z_ω is not set as p^ω in our experiments.

To detect cycling, we compare values of h_i in consecutive iterations. Since the continuous values of hash weights and Lagrangian multipliers may result in non-integer hash values, we compare the hash values of patients in consecutive iterations by using a small threshold value. If the difference between the hash values are smaller than the threshold value over three consecutive iterations, it is considered as the indicator of a cycle. The cycle then needs to be broken to maintain algorithm convergence.

After a cycle is detected, the appointment time for patient i can be fixed to a certain value. There might be different approaches for fixing the appointment time, which include the minimum and maximum appointment times observed among all scenario subproblem solutions for that patient in the previous iteration. Based on our preliminary experiments, we fix the appointment time to the one that is most repeatedly observed among subproblem solutions, which we call the *majority value*. We use this approach after the first 50 iterations of the PHA, since there is high variation in the appointment times of a patient across different scenarios in the beginning iterations. The coupled implementation of cycle detection and variable fixing methods ensures the convergence of the PHA to a local optimum in finitely many steps.

A common observation in our preliminary experiments is that for many patients, the appointment times for a majority of the scenarios converge quickly, whereas those of the remaining scenarios fail to converge for many iterations. To overcome this, we fix the appointment time of a patient if there is convergence in 80% of the scenarios. Although this may reduce the solution quality, it significantly reduces computational times.

4.3.4 Varying CPLEX optimality gap value in the subproblems

In the earlier iterations of the PHA, computation times to solve the subproblems are longer, since the algorithm uses lower values of penalty parameter ρ. However, the PHA does not need optimal solutions of the subproblems for convergence. We may heuristically solve the subproblems to decrease solution times. For this purpose, we test the idea proposed by Watson and Woodruff (2011), where the CPLEX optimality gap value for the subproblems is varied according to the convergence behaviour. In particular, it is monotonically decreased by checking the following

condition:

$$gap_{v+1} = \min \left\{ \sum_{i \in I, \hat{a_i}^{(v)} \neq 0} \sum_{\omega \in \Omega} \frac{|a_i^{\omega(v)} - \hat{a_i}^{(v)}|}{\hat{a_i}^{(v)}},\ gap_v \right\} \tag{35}$$

The gap value is decreased if the appointment times of the patients move closer to the consensus parameter. Otherwise, we keep the gap value equal to that of the previous iteration. The initial gap value is determined according to the results of the preliminary experiments.

Using approximate solutions within the PHA may either improve solution quality by making greater amount of iterations or reduce it because of the low-quality solutions particularly at the earlier iterations. The details of this procedure and results are discussed in Section 5.

4.3.5 Lower Bounds on Total Nurse Overtime and Chair Idle Time

In the first iteration of the PHA, the scenario subproblems are solved without adding any Lagrangian or quadratic terms. Since the durations are known in each subproblem, waiting times for all patients would be reduced to zero without causing an increase in the total overtime or idle time values. In other words, the trade-off between waiting time and the sum of other measures disappears as the non-anticipativity constraint is totally omitted under this setting. Then, nurse overtime and chair idle time remain as the challenging terms in the subproblem objective function. Note that these two measures do not conflict with each other. Therefore, the resulting sum of nurse overtime and chair idle time values in a scenario subproblem solution in the first iteration constitutes a lower bound for the associated scenario subproblem in the subsequent iterations.

5 Computational Experiments

In this section, we discuss the computational experiments for the Chemotherapy Appointment Scheduling Problem. We test the methods using data gathered from the Outpatient Chemotherapy Unit at Hacettepe Oncology Hospital from November 2017 to March 2018. The data set includes realized pre-medication and infusion times as well as estimated treatment durations for all patients.

In our experiments, Microsoft Visual C++ 2019 is used for the implementation of the algorithm using CPLEX 12.9 Concert Technology. The computations are performed with Intel (R) Core (TM) i7-9750H CPU @2.60 GHz and 16GB RAM. We conduct each experiment on a problem instance set that consists of 10 instances. We generate those problem instances by sampling pre-medication

and infusion durations in our data set.

In Section 5.1, the descriptive statistics on the data set is provided and the problem instance generation approach is discussed. The method proposed to improve scenario subproblem solution times is tested in Section 5.2. The performance of the LPHA is assessed in Section 5.3, whereas sensitivity analysis on model parameters is conducted to generate managerial insights in Section 5.4. Finally, the value of stochastic solution (VSS) is estimated in Section 5.5.

5.1 Data Description and Problem Instance Generation

Our data set consists of the estimated and actual pre-medication and treatment durations of 204 patients we have collected data on over 11 different days. The total treatment duration for these patients ranges between 16 and 240 minutes. In particular, pre-medication duration varies between 0 and 36 minutes with the average of 15.3 minutes. Infusion duration changes between 16 and 217 minutes, and the average is 112.5 minutes. Furthermore, a 95% confidence interval for the means of pre-medication and infusion durations are [14.61, 16.09] and [104.56, 120.35] minutes, respectively.

We next compare the realized treatment durations with the estimated durations. The data set indicates that 59% of the patients were allocated less time than needed, whereas the remaining 41% were allocated more than the necessary amount. Furthermore, the distribution of patient groups (by estimated duration) varies significantly among different half-shifts. To avoid sampling bias due to this reason, we generate representative half-shifts in our instances, rather than replicating any of the half-shifts. In Figure 2, the comparison of actual and predicted treatment times can be observed on a scatter plot. In horizontal and vertical axes, the predicted and actual treatment times are provided, respectively. The value of mean absolute percent error (MAPE) is 16.9%, which shows a significant variability in observed treatment time values compared to predicted values. Furthermore, Figure 2 also indicates an underestimation of the actual times of treatment times shorter than 150 minutes, and overestimation of those longer than three hours. Therefore, the manager of the chemotherapy clinic should consider a scheduling model where the variability of treatment times are taken into account.

In the Hacettepe Outpatient Chemotherapy Unit, the schedules are designed for half-day periods. To mimic the current practice followed in the unit, we also design half-day schedules. We generate each problem instance by sampling pre-medication and infusion durations from the data set. As can also be seen in Figure 2, there are 21 discrete groups based on the estimated treatment duration of the patient. Since each half-shift consists of 7–8 patients and some of the 21 groups have

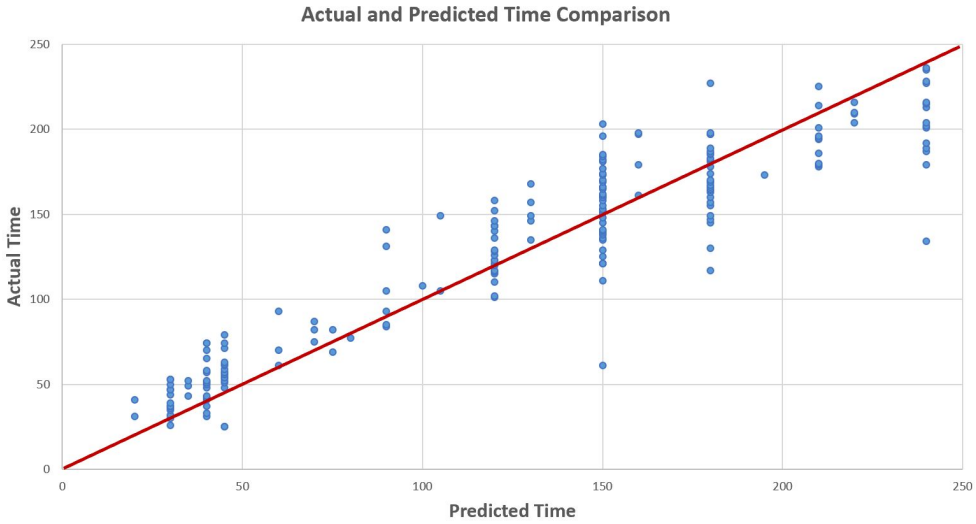

Figure 2: Comparison of actual and predicted treatment times

very limited number of observations, we cluster these into four larger classes with cut-offs of 45, 100, 150 and 240 minutes of estimated treatment time. For each class, the actual pre-medication and infusion durations are analyzed. The intervals for the realized pre-medication and infusion durations for the defined classes as well as the probability of observing the defined classes in the data set are provided in Table 2.

INSERT TABLE 2 HERE

To generate the treatment duration for a patient, we follow the following procedure: We first draw a random number between 0 and 1 and use the associated probabilities in Table 2 in order to determine the patient's class. Next, pre-medication and infusion durations of this patient are determined by assuming a uniform distribution within the actual pre-medication and infusion intervals in Table 2 for her class. To justify the use of the uniform distribution for these times, we provide the chi-square goodness-of-fit test results and plots of fitted vs. actual durations in Appendix C. We generate a different set of durations for this patient in each scenario by taking into account her class. We then repeat this process for each patient in each instance. In generating these instances, we also aim to ensure that the total estimated treatment time results in an expected overtime around 60 minutes per nurse, in line with the requirements of the unit. For each instance, we set the overtime limit L large enough to avoid infeasibility, but also small enough to avoid unfair nurse schedules. In our experiments, the overtime value for any nurse in any scenario does not exceed 147 minutes. Furthermore, none of our instances results in zero overtime, which is another indication that our instances are generated in line with the actual assignment of patients to half-shifts.

Our instances are labeled as i_j_k, where i, j, and k refer to the instance number, number of patients, and number of scenarios, respectively. Note that we provide the explicit composition of patient classes considered in each of the 10 instances in Appendix D.

5.2 Tests for varying CPLEX optimality gap value in the subproblems

A technique for solving scenario subproblems approximately based on optimality gap values that vary over iterations was introduced in Section 4.3.4. The optimality gap values exhibit monotonically nonincreasing behavior over iterations due to the given rule. We perform experiments to see the effects of this technique on the computational times and quality of solutions by starting with two different initial optimality gap values of 10% and 15%. The results of the experiments are presented in Table 3. In these experiments, the number of patients, nurses and chairs are set as 8, 2, and 4, respectively.

INSERT TABLE 3 HERE

The results are compared with the version of the LPHA that uses default CPLEX optimality gap value while solving subproblems. There is no guarantee that the overall computation times decrease when the optimality gap is increased, because the algorithm may converge after larger amount of iterations. Indeed, the one that uses default CPLEX optimality gap value performs the best in terms of solution quality and run time on average. Due to insufficient basis for the use of rule, we do not vary the optimality gap in our further experiments.

We also test various parameters of the LPHA to determine the best values to use in further experiments. The details of these experiments are provided Appendix E. Consequently, we set the step size α as 2, penalty parameter ρ as 0.0001, upper bound ρ_1^u for the penalty parameter as 0.1, and the iteration limit as 100.

5.3 Assessment of the LPHA Performance

In this section, we initially evaluate the performance of the LPHA by conducting comparisons with optimal solutions. Next, the LPHA solutions are compared with the baseline schedule (i.e., actual schedule of the particular outpatient chemotherapy unit at Hacettepe Oncology Hospital). Finally, the LPHA is tested against commonly used scheduling heuristics from the literature.

5.3.1 Comparison with Optimal Solutions

To validate the performance of the LPHA, we first compare our solutions to optimal solutions for instances that can be solved to optimality by CPLEX. Since CPLEX cannot find the optimal solutions in three hours of time limit even for problem instances having 8 patients, 5 scenarios, 2 nurses and 4 chairs, we decrease the number of patients to 7 for these experiments. We use five different λ values on 10 different instances.

On average, CPLEX spends approximately 100 seconds to find the optimal solution, whereas the LPHA spends approximately 40 seconds. The average optimality gap for the LPHA solutions is 5.64%, which shows that the LPHA finds high quality solutions within significantly lower computational times than CPLEX.

Using these instances, we are also able to observe the structure of optimal solutions. For this end, Figure 3 shows the optimal chair and nurse schedules for the first two scenarios of the first instance in this set. One common theme in both solutions is the tight scheduling of appointment times in the beginning of the shift until all chairs are occupied. The remaining patients are scheduled to arrive in sequence based on the expected departure times of the patients currently receiving treatment, with some buffer time to avoid waiting. Indeed, scenario 1 results in only 2 minutes of patient waiting (for patient 2), whereas there is no patient waiting in scenario 2. Scenario 1 also results in no overtime, as patients arriving towards the end of the shift need less time for treatment compared to scenario 2, where 7 minutes of overtime is needed for nurse 2 to complete the treatment of patient 3.

From Figure 3, we also observe the importance of assigning chairs and nurses to patients upon arrival, rather than a priori. For example, fixing the chair assignment of patients 5, 6, and 7 over all possible scenarios would result in either waiting of patients or nurse overtime, unless more resources are added. These scenarios also reveal the importance of relaxing the slot-based scheduling structure, since hourly or 2-hourly slots would lead to more patient waiting time and/or more nurse overtime in both scenarios.

5.3.2 Comparison with the Baseline Schedule

To demonstrate the benefit of using LPHA to create schedules at the Outpatient Chemotherapy Unit at Hacettepe Oncology Hospital, we compare the LPHA solutions with the baseline schedule. Since the LPHA solves instances that represent half-day shifts, we generate two schedules using our

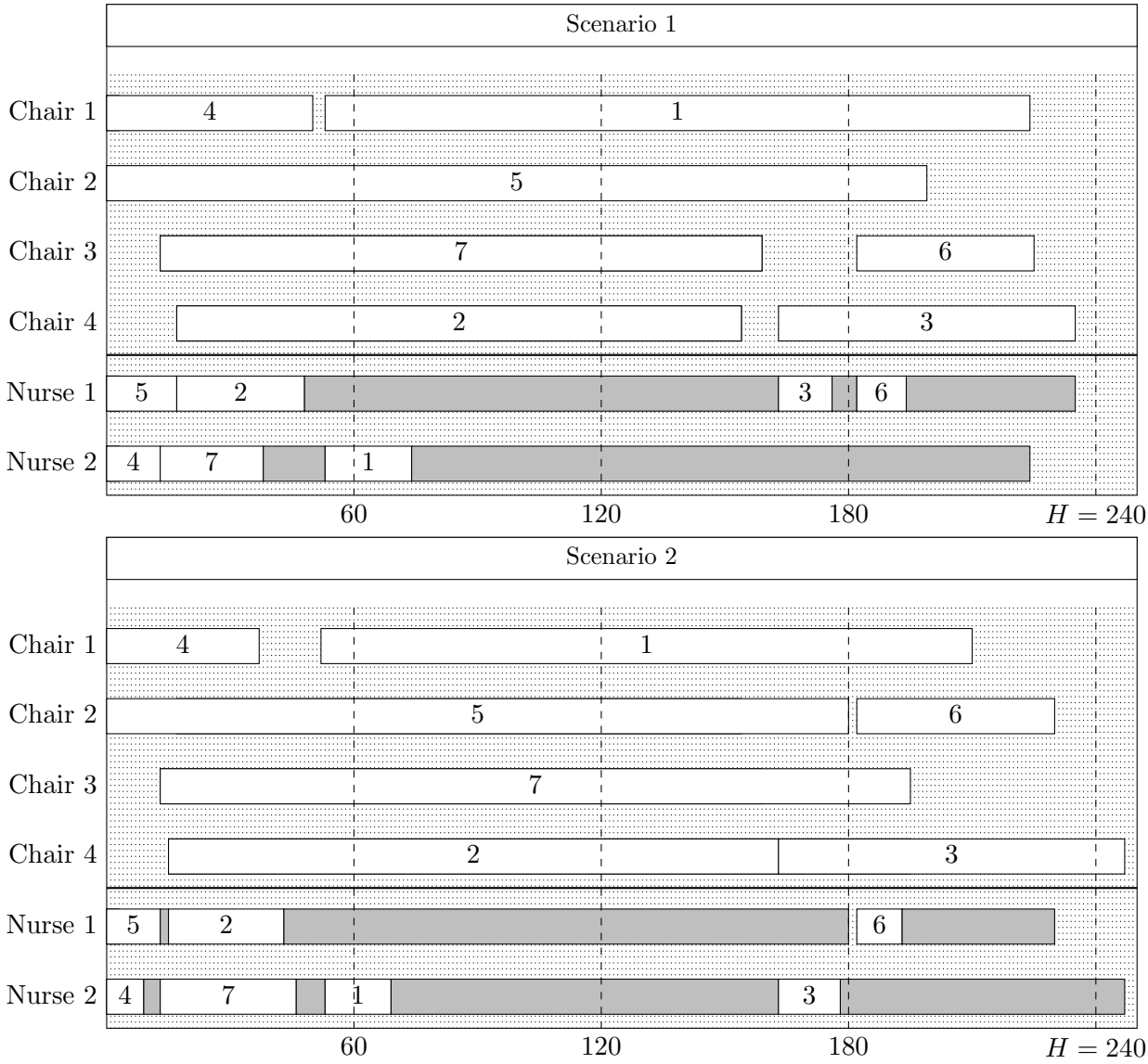

Figure 3: Optimal 4-hour chair and nurse schedules for two scenarios of Instance 1, where each number corresponds to a patient and the appointment times of patients are $a_1 = 52$, $a_2 = 16$, $a_3 = 163$, $a_4 = a_5 = 0$, $a_6 = 182$, and $a_7 = 13$

algorithm for a representative day in our data set. The resulting morning and afternoon schedules are compared with the corresponding shifts of the baseline schedule for the same day on Table 4. Note that we set $\lambda_1, \lambda_2, \lambda_3$ values as 0.3, 0.3, 0.4, respectively. The number of patients, nurses and chairs are fixed as 8, 2, and 4 in these experiments, respectively.

While selecting the day for schedule comparison, we consider the percentage of patient classes observed in our data set, which is given on Table 2 . We then identify a day in which the patient class composition is consistent with those average values. It should be noted here that the fixed-time slots used by the Outpatient Chemotherapy Unit start at 8:00 and 10:30 for the morning shift, and 13:00 and 15:30 for the afternoon shift. Since the chair set considered in each instance consists of four chairs and eight patients are scheduled for the set at each shift, four patients are asked to arrive at the beginning of each time slot in the baseline schedule. The LPHA schedule is also generated based on the same patient class composition. Then, using the same scenario set for treatment durations, the baseline schedule is compared with the LPHA schedule. To find the performance measures associated with the baseline schedule for each shift, the second-stage problem of the SMIP is solved.

Table 4 compares the LPHA and baseline schedules with respect to average waiting time, nurse overtime and chair idle time. All three measures are improved by the LPHA at both shifts. The use of fixed-length slots at the Outpatient Chemotherapy Unit explains the reason behind the comprehensive improvement. The scope of improvement is particularly large with respect to average waiting time. The tendency to underestimate the majority of the appointment durations in the unit results in excessive wait amounts for patients. Note that the rate of reduction of nurse overtime is also notable, particularly for the morning shift.

INSERT TABLE 4 HERE

5.3.3 Comparison with Scheduling Heuristics

To compare the performance of the LPHA with practice in the other outpatient chemotherapy units, we assess its performance against various combinations of sequencing and appointment time setting heuristics from the relevant literature. We sequence patients using four different sequencing heuristics: increasing mean of treatment time (SPT), decreasing mean of treatment time (LPT), increasing variance of treatment time (VAR), and increasing coefficient of variation of treatment time (CoV). We consider a *job hedging* heuristic to determine the estimated lengths of patient

appointments.

Before discussing the implementation details of the heuristics, the structure of the model in terms of first and second-stage decisions is investigated. When values for patient precedence b_{ij} and appointment time a_i variables become known in the first-stage, it is then easy to settle on the second-stage decisions. In some appointment scheduling studies, such as Mancilla and Storer (2012), the second-stage subproblems are considered as network flow problems. Having multiple resources in our case makes the subproblems more challenging. However, the optimal nurse and chair assignment decisions at this stage can still be made easily using a simple rule that checks b_{ij} values. The rule would favor the patient preceding others in the list while assigning the first available nurse and chair. Next, actual treatment start time for each patient can be determined accordingly, and their waiting times are revealed according to the discharge time of the previously scheduled patients. By checking discharge times of the patients treated by a nurse, overtime value for each nurse can be easily calculated. Since solving the second-stage problem is easy, finding a heuristic approach to solve the first-stage problem is a more critical issue. After setting the first-stage variable values, we call CPLEX rather then using the simple rule to solve the second-stage problem, as CPLEX finds the optimal solution within a very short amount of time.

To create schedules, we first choose one of the four sequencing heuristics discussed above. There are two different durations that may affect the sequencing rules, namely average pre-medication and infusion durations. We use average treatment time, which is equal to the summation of those two values.

After the sequence is fixed, we utilize the job hedging heuristic to set patient appointment times (Yellig and Mackulak, 1997). We apply the heuristic in accordance with the approach also used in Gul (2018), Castaing et al. (2016), and Gul et al. (2011). We sort the durations for the relevant patient class in the data set in non-increasing order, and calculate the k^{th} percentile of the set to use it for determining the estimated treatment duration of a patient. Since there are two different chemotherapy durations in the problem, the percentile values associated with them are separately determined, and the summation of those two values is considered while setting patient appointment times. The steps of the scheduling heuristic that combines job hedging with one of the sequencing heuristics, namely LPT, is demonstrated in Algorithm 3.

We vary the percentile values between 40% and 65% in increments of 5% for all sequencing rules and fix the values of λ_1, λ_2, and λ_3 as 0.1, 0.8, and 0.1, respectively. Table 5 compares the performances of the heuristics with that of the LPHA by illustrating the average gap between the

Algorithm 3 Job Hedging Heuristics for LPT Rule

1: Calculate the average treatment duration for every patient using $avg_i = \frac{\sum_{\omega \in \Omega}(s_i^\omega + t_i^\omega)}{|\Omega|} \quad \forall i \in I.$
2: Sort the values of the avg_i according to LPT rule, assign index numbers to patients and sequence the patients according to their index numbers.
3: Assign b_{ij} values according to index numbers found at Step 2.
4: Set appointment times according to given percentile level of job hedging heuristic. If the index of the patient is less than or equal to the both of the numbers of chair and nurse, the appointment time of the associated patient is assigned as zero. The appointment times of the other patients are set sequentially according to their index numbers by checking the first estimated available time of nurse and chair simultaneously.
5: Call CPLEX to obtain second-stage decisions and objective function value.

solutions of the LPHA and each combination of sequencing and job hedging heuristic for a specific percentile level.

INSERT TABLE 5 HERE

The results show that the LPHA outperforms all combinations of heuristics significantly. On the average, LPT rule performs better than the other sequencing rules. This result makes sense since the patients with longer treatment durations are assigned to the earlier hours of the day also at the outpatient chemotherapy unit in the Hacettepe Oncology Hospital. Note that the LPHA improves the solutions of even the best-performing and commonly used rule by 37.7%. Furthermore, the percentiles 45%, 40%, 50%, and 40% used for assigning appointment times are the best in terms of solution quality on average for sequencing rules SPT, LPT, VAR, and CoV, respectively.

We also compare the LPHA with commonly used sequencing rules without using job hedging. The sequence of patients is fixed with one of the four aforementioned sequencing rules. Then, CPLEX is called to determine the appointment times and nurse/chair assignments. We call these modified versions of the heuristics as *SPT-opt, LPT-opt, VAR-opt*, and *CoV-opt*, respectively. Although we fix the sequence of the patients, assigning appointment times to the patients still remains a challenging optimization problem for CPLEX due to the uncertainty of the treatment durations. Therefore, we put a three-hour time limit on CPLEX and report the best solution for the given sequence. In Table 6, we compare the results in terms of average gap between the solutions of the LPHA and each type of sequencing rules.

INSERT TABLE 6 HERE

As shown in Table 6, the LPHA outperforms all heuristics, even when the appointment times and second-stage decisions are determined by a mathematical model, rather than simple job hedging

rules. LPT-opt, which outperforms all other heuristics, performs 11.9% worse on average than the LPHA in terms of solution quality.

INSERT TABLE 7 HERE

To further investigate the differences in the solutions between the LPHA and benchmark heuristics, Table 7 provides the appointment times for the eight patients, expected waiting time, overtime, chair idle time, and objective value of the instance 1_8_50 under various heuristic approaches. Comparison of the LPHA appointment times to those under LPT-opt shows that the order of the patients in both cases is very similar and the differences between the appointment times are within 17 minutes. The LPHA sacrifices some waiting time to improve the nurse overtime, for which the objective weight is higher. The differences in the appointment times become more significant as job hedging heuristics are used and percentile values are increased, which explains the difference between the objective values. Under SPT, the order of patients is almost completely reversed from that under the LPHA. This results in shorter expected patient waiting times, but substantially increases the nurse overtime, leading to the inferiority of the SPT-based heuristics to the LPT-based counterparts and the LPHA.

Note that the relative performances of sequencing rules shown on Table 6 are consistent with the results of Table 5. Even though the average gaps improve when CPLEX is used to set appointment times instead of job hedging heuristics, the ranks of the sequencing rules do not change. Therefore, the LPT would be preferred by the managers creating schedules without using any sophisticated tool. However, the most important finding of this section is that the managers need a sophisticated approach rather than simple rules to determine patient sequences.

5.4 Sensitivity Analysis on Model Parameters

In this section, we present a sensitivity analysis on the model parameters to generate managerial insights. The value of λ and number of nurses and chairs are varied for this purpose.

5.4.1 Impact of the λ value

The chemotherapy unit manager would typically consider the trade-off between patient waiting time, nurse overtime, and chair idle time while designing schedules. To illustrate the scope of this trade-off, we vary the λ values, where (λ_1,λ_2,λ_3) are the objective function coefficients associated with patient waiting time, nurse overtime, and chair idle time, respectively. λ values equal to zero

refers to the extreme case where the related objective term is not important performance measure for the model. We test twelve set of different values of λ on an instance set. We fix the number of patients, nurses, and chairs to 8, 2, and 4, respectively in these experiments.

We set the λ values systematically across experiments. While the weights for any two measures are kept the same, the weight of the third measure may be (i) one-tenth, (ii) one-half of the others, or (iii) twice, (iv) ten times as much as the others in an experiment. After fixing the weight of waiting time as 1, our weight-setting procedure results in the weight combinations shown in the first column of Table 8. Note that we normalized the values of λ in the experiments so that they could be consistent with the procedure in the other experiments.

INSERT TABLE 8 HERE

Table 8 clearly illustrates the trade-off between patient waiting time, nurse overtime, and chair idle time with reference to λ values. Average patient waiting time decreases significantly when only the relative value of λ_1 is increased. Average nurse overtime and chair idle time values also drop respectively as each of the λ_2 and λ_3 values is increased individually while keeping the remaining two constant. However, the impact of λ_2 and λ_3 on their respective performance measures is not as dramatic as the impact of λ_1 on average waiting time.

When only the λ_1 value increases, both average nurse overtime and chair idle time increase. To minimize patient waiting time, the model assigns appointment times in wider time intervals, which leads to increased overtime and idle time values. This shows that there exists trade-off between patient waiting time and the sum of nurse overtime and chair idle time. When only the λ_2 is increased, average waiting time increases, but average idle time decreases. This indicates that minimizing overtime also helps to minimize chair idle time, meaning that nurse overtime and chair idle time are not conflicting measures. When only the λ_3 value is increased, average waiting time increases, while the average nurse overtime stays about the same. This implies that increasing the λ_3 value alone may not always help improve average nurse overtime. Note that the cases where the λ values are 0 and 1 are not realistic for the unit manager. The manager may choose a λ value between those extreme points according to the goals of the units.

5.4.2 Impact of the number of nurses and chairs

The decision maker in the chemotherapy unit would also be interested in investigating the effect of using different numbers of chairs and nurses on the performance measures. For this purpose, we

vary the number of nurses and chairs in our instance sets between 1 and 3, and 4 and 6, respectively. We ignore the case with 3 nurses and 4 chairs, since the corresponding nurse to chair ratio is not realistic. We report the average objective values, run times, and performance measures in the objective function based on 10 instances in Tables 9 and 10.

INSERT TABLE 9 HERE

INSERT TABLE 10 HERE

As expected, when the number of nurses is increased, the average objective value decreases. All performance measures in the objective function (patient waiting time, nurse overtime, and chair idle time) are improved. The increase in the number of nurses provides to have more flexible schedules without extending clinic closure time. Therefore, both nurse overtime and chair idle time reduces. On the other hand, increase in the number of chairs results with having higher chair idle times. Therefore, the objective function value might increase depending on the net change in the decrease of patient waiting time and nurse overtime and increase in the chair idle time. Another interesting issue is the change in the computation times. An increase in the number of chairs and nurses generally makes subproblems easier to solve resulting with a reduction in the solution times.

5.5 Estimating the Value of Stochastic Solution (VSS)

VSS is a cost of excluding uncertainty while giving the first-stage decisions. It measures the benefit of using stochastic programming solution over the mean value solution. We estimate the VSS by calculating the difference between the expected objective value associated with the mean value solution and the LPHA solution. In Table 11, we report the relative VSS, which represents the percentage change in the objective values for various λ values.

INSERT TABLE 11 HERE

As expected, VSS is always positive in our experimental runs. The LPHA improves the solutions of the mean value problem by 27.9% on average. Therefore, considering uncertainty in chemotherapy appointment scheduling problem is valuable to minimize total weighted sum of patient waiting time, nurse overtime, and chair idle time. Note that the benefit of uncertainty modeling is particularly significant (with VSS% of 58.8%) for the cases where the highest priority is assigned to patient waiting time.

6 Conclusion

Over the recent years, due to the increase in the prevalence of cancer, the demand for chemotherapy units has been growing, and these units should have efficient planning and scheduling structures. In this paper, we focused on the Chemotherapy Appointment Scheduling Problem, where patients are sequenced and assigned appointment times by considering the availability of nurses and infusion chairs at the same time. The uncertainty in both the pre-medication and infusion durations are considered in the study. The aim is to design an appointment schedule in order to minimize the expected weighted sum of patient waiting time, nurse overtime, and chair idle time. A two-stage stochastic mixed integer programming formulation is used to formulate the problem.

The uncertain durations of chemotherapy are represented with scenarios in the stochastic programming formulation. Solving the problem even with a very few scenarios with CPLEX is computationally challenging. Therefore, a scenario decomposition based algorithm, namely the progressive hedging algorithm, is implemented to solve the problem. The PHA is enhanced through a number of modifications. A penalty parameter update method is proposed, where the parameters are set dynamically and controlled using changing limits. A cycle detection method is used to guarantee convergence of the algorithm. A variable fixing procedure is incorporated to reduce overall computation times. Subproblem solution times are improved through symmetry-breaking constraints and bounds imposed on nurse overtime in the constraint set, and by linearization of the quadratic term in the objective function. The resulting algorithm after enhancements is called as linearized PHA (LPHA).

The proposed method is implemented based on real data from the Hacettepe Outpatient Chemotherapy Unit. To evaluate the performance of algorithm, the LPHA solutions are compared with the CPLEX solutions and the baseline schedule. Moreover, several scheduling heuristics are used to validate the performance of the algorithm. The results show that the LPHA outperforms commonly used heuristics in all cases. This finding implies that a sophisticated approach rather than simple rules is necessary to determine patient sequences in outpatient chemotherapy units. Furthermore, VSS is estimated to assess the value of considering uncertainty in our problem. It is found that the LPHA improves the solutions of mean value problem significantly. Our solution approach can be useful for chemotherapy unit managers to evaluate the effects of appointment schedules on nurse overtime, chair idle time, and patient waiting time. The unit managers can also observe the effects of changing the level of nurses and chairs in the chemotherapy unit.

In a future study, the infusion schedules can be coordinated with lab test and oncologist evaluation schedules. A comprehensive model can be developed to determine treatment days and appointment times for patients simultaneously. Scenario subproblem solution routine in the LPHA can be parallelized to reduce computational time and solve larger size problem instances. Making the assumption of preserving patient arrival sequence throughout the whole chemotherapy unit visit allowed us to formulate the problem as a two-stage model. A multi-stage stochastic programming model can be formulated to study a variant of our problem by relaxing this assumption.

References

[1] Amir Ahmadi-Javid, Zahra Jalali, and Kenneth J Klassen. Outpatient appointment systems in healthcare: A review of optimization studies. *European Journal of Operational Research*, 258(1):3–34, 2017.

[2] Shabbir Ahmed. A scenario decomposition algorithm for 0–1 stochastic programs. *Operations Research Letters*, 41(6):565–569, 2013.

[3] Michelle Alvarado and Lewis Ntaimo. Chemotherapy appointment scheduling under uncertainty using mean-risk stochastic integer programming. *Health Care Management Science*, 21(1):87–104, 2018.

[4] Hyun-Jung Alvarez-Oh, Hari Balasubramanian, Ekin Koker, and Ana Muriel. Stochastic appointment scheduling in a team primary care practice with two flexible nurses and two dedicated providers. *Service Science*, 10(3):241–260, 2018.

[5] Semih Atakan and Suvrajeet Sen. A progressive hedging based branch-and-bound algorithm for mixed-integer stochastic programs. *Computational Management Science*, pages 1–40, 2018.

[6] Sakine Batun, Brian T. Denton, Todd R. Huschka, and Andrew J. Schaefer. Operating room pooling and parallel surgery processing under uncertainty. *INFORMS Journal on Computing*, 23(2):220–237, 2011.

[7] Mehmet Begen and Maurice Queyranne. Appointment scheduling with discrete random durations. *Mathematics of Operations Research*, 36(2):240–257, 2011.

[8] Bjorn P. Berg, Brian T. Denton, S. Ayca Erdogan, and Thomas Rohleder. Optimal booking and scheduling in outpatient procedure centers. *Computers and Operations Research*, 50:24–37, 2014.

[9] John R Birge and Francois Louveaux. *Introduction to stochastic programming*. Springer Science & Business Media, 2011.

[10] Cancer Research UK. Worldwide cancer statistics. https://www.cancerresearchuk.org/health-professional/cancer-statistics, 2018.

[11] Jeremy Castaing, Amy Cohn, Brian T Denton, and Alon Weizer. A stochastic programming approach to reduce patient wait times and overtime in an outpatient infusion center. *IIE Transactions on Healthcare Systems Engineering*, 6(3):111–125, 2016.

[12] Tugba Cayirli and Emre Veral. Outpatient scheduling in health care: a review of literature. *Production and Operations Management*, 12(4):519–549, 2003.

[13] Kwok Cheung, Dinakar Gade, César Silva-Monroy, Sarah M Ryan, Jean-Paul Watson, Roger J-B Wets, and David L Woodruff. Toward scalable stochastic unit commitment. *Energy Systems*, 6(3):417–438, 2015.

[14] Teodor Gabriel Crainic, Xiaorui Fu, Michel Gendreau, Walter Rei, and Stein W Wallace. Progressive hedging-based metaheuristics for stochastic network design. *Networks*, 58(2):114–124, 2011.

[15] Brian Denton and Diwakar Gupta. A sequential bounding approach for optimal appointment scheduling. *IIE Transactions*, 35(11):1003–1016, 2003.

[16] Brian Denton, James Viapiano, and Andrea Vogl. Optimization of surgery sequencing and scheduling decisions under uncertainty. *Health care management science*, 10(1):13–24, 2007.

[17] Roxanne Dobish. Next-day chemotherapy scheduling: a multidisciplinary approach to solving workload issues in a tertiary oncology center. *Journal of Oncology Pharmacy Practice*, 9(1):37–42, 2003.

[18] Dinakar Gade, Gabriel Hackebeil, Sarah M Ryan, Jean-Paul Watson, Roger J-B Wets, and David L Woodruff. Obtaining lower bounds from the progressive hedging algorithm for stochastic mixed-integer programs. *Mathematical Programming*, 157(1):47–67, 2016.

[19] Raphael EC Gonçalves, Erlon Cristian Finardi, and Edson Luiz da Silva. Applying different decomposition schemes using the progressive hedging algorithm to the operation planning problem of a hydrothermal system. *Electric Power Systems Research*, 83(1):19–27, 2012.

[20] Serhat Gul. A stochastic programming approach for appointment scheduling under limited availability of surgery turnover teams. *Service Science*, 10(3):277–288, 2018.

[21] Serhat Gul, Brian T Denton, and John W Fowler. A progressive hedging approach for surgery planning under uncertainty. *INFORMS Journal on Computing*, 27(4):755–772, 2015.

[22] Serhat Gul, Brian T Denton, John W Fowler, and Todd Huschka. Bi-criteria scheduling of surgical services for an outpatient procedure center. *Production and Operations Management*, 20(3):406–417, 2011.

[23] Diwakar Gupta and Brian Denton. Appointment scheduling in health care: Challenges and opportunities. *IIE transactions*, 40(9):800–819, 2008.

[24] Shoshana Hahn-Goldberg, J Christopher Beck, Michael W Carter, Maureen Trudeau, Philomena Sousa, and Kathy Beattie. Solving the chemotherapy outpatient scheduling problem with constraint programming. *Journal of Applied Operational Research*, 6(3):135–144, 2014.

[25] Thorkell Helgason and Stein W Wallace. Approximate scenario solutions in the progressive hedging algorithm. *Annals of Operations Research*, 31(1):425–444, 1991.

[26] Arild Helseth. Stochastic network constrained hydro-thermal scheduling using a linearized progressive hedging algorithm. *Energy Systems*, 7(4):585–600, 2016.

[27] Alireza F Hesaraki, Nico P Dellaert, and Ton de Kok. Generating outpatient chemotherapy appointment templates with balanced flowtime and makespan. *European Journal of Operational Research*, 275(1):304–318, 2019.

[28] M Heshmat, K Nakata, and A Eltawil. Solving the patient appointment scheduling problem in outpatient chemotherapy clinics using clustering and mathematical programming. *Computers & Industrial Engineering*, 124:347–358, 2018.

[29] Anali Huggins, David Claudio, and Eduardo Pérez. Improving resource utilization in a cancer clinic: an optimization model. In *Proceedings of the 2014 Industrial and Systems Engineering Research Conference*, page 1444, 2014.

[30] Youngbum Hur, Jonathan F. Bard, and Douglas J. Morrice. Appointment scheduling at a multidisciplinary outpatient clinic using stochastic programming. *Naval Research Logistics*, 2020.

[31] Lars Magnus Hvattum and Arne Løkketangen. Using scenario trees and progressive hedging for stochastic inventory routing problems. *Journal of Heuristics*, 15(6):527, 2009.

[32] Kenneth J. Klassen and Reena Yoogalingam. Appointment scheduling in multi-stage outpatient clinics. *Health Care Management Science*, 22:229–244, 2019.

[33] Qingxia Kong, Shan Li, Nan Liu, Chung-Piaw Teo, and Zhenzhen Yan. Appointment scheduling under time-dependent patient no-show behavior. *Management Science*, forthcoming, 2019.

[34] G. Lamè, O. Jouini, and Julie Stal-Le Cardinal. Outpatient chemotherapy planning: A literature review with insights from a case study. *IIE Transactions on Healthcare Systems Engineering*, 6(3):127–139, 2016.

[35] Gilbert Laporte and François V Louveaux. The integer l-shaped method for stochastic integer programs with complete recourse. *Operations Research Letters*, 13(3):133–142, 1993.

[36] A.G. Leeftink, I.M.H. Vliegen, and E.W. Hans. Stochastic integer programming for multidisciplinary outpatient clinic planning. *Health Care Management Science*, 22:53–67, 2019.

[37] Bohui Liang and Ayten Turkcan. Acuity-based nurse assignment and patient scheduling in oncology clinics. *Health Care Management Science*, 19(3):207–226, 2016.

[38] Bohui Liang, Ayten Turkcan, Mehmet Erkan Ceyhan, and Keith Stuart. Improvement of chemotherapy patient flow and scheduling in an outpatient oncology clinic. *International Journal of Production Research*, 53(24):7177–7190, 2015.

[39] Ho-Yin Mak, Ying Rong, and Jiawei Zhang. Sequencing appointments for service systems using inventory approximations. *Manufacturing and Service Operations Management*, 16(2):251–262, 2014.

[40] Camilo Mancilla and Robert Storer. A sample average approximation approach to stochastic appointment sequencing and scheduling. *IIE Transactions*, 44(8):655–670, 2012.

[41] Avishai Mandelbaum, Petar Momcilovic, Nikolaos Trichakis, Sarah Kadish, Ryan Leib, and Craig A Bunnell. Data-driven appointment-scheduling under uncertainty: The case of an infusion unit in a cancer center. *Management Science*, 66(1):1–28, 2019.

[42] John M Mulvey and Hercules Vladimirou. Applying the progressive hedging algorithm to stochastic generalized networks. *Annals of Operations Research*, 31(1):399–424, 1991.

[43] National Cancer Institute. Cancer Statistics. https://www.cancer.gov/about-cancer/understanding/statistics, 2018. Accessed: November 2018.

[44] Eduardo Perez, Lewis Ntaimo, Cesar O. Malave, Carla Bailey, and Peter McCormack. Stochastic online appointment scheduling of multi-step sequential procedures in nuclear medicine. *Health Care Management Science*, 16(4):281–299, 2013.

[45] Atle Riise, Carlo Mannino, and Leonardo Lamorgese. Recursive logic-based benders' decomposition for multi-mode outpatient scheduling. *European Journal of Operational Research*, 255(3):719–728, 2016.

[46] Hannah Ritchie and Max Roser. Causes of death. https://ourworldindata.org/causes-of-death, 2018. Accessed: November 2018.

[47] R Tyrrell Rockafellar and Roger J-B Wets. Scenarios and policy aggregation in optimization under uncertainty. *Mathematics of Operations Research*, 16(1):119–147, 1991.

[48] Abdellah Sadki, Xiaolan Xie, and Franck Chauvin. Appointment scheduling of oncology outpatients. In *2011 IEEE International Conference on Automation Science and Engineering*, pages 513–518. IEEE, 2011.

[49] Pablo Santibáñez, Ruben Aristizabal, Martin L Puterman, Vincent S Chow, Wenhai Huang, Christian Kollmannsberger, Travis Nordin, Nancy Runzer, and Scott Tyldesley. Operations research methods improve chemotherapy patient appointment scheduling. *The Joint Commission Journal on Quality and Patient Safety*, 38(12):541–AP2, 2012.

[50] Suvrajeet Sen and Julia L Higle. The c3 theorem and a d2 algorithm for large scale stochastic mixed-integer programming: set convexification. *Mathematical Programming*, 104(1):1–20, 2005.

[51] Suleyman Sevinc, Ulus Ali Sanli, and Erdem Goker. Algorithms for scheduling of chemotherapy plans. *Computers in Biology and Medicine*, 43(12):2103–2109, 2013.

[52] Takahiro Tanaka. *Infusion chair scheduling algorithms based on bin-packing heuristics*. State University of New York at Binghamton, 2011.

[53] Ayten Turkcan, Bo Zeng, and Mark Lawley. Chemotherapy operations planning and scheduling. *IIE Transactions on Healthcare Systems Engineering*, 2(1):31–49, 2012.

[54] Richard M Van Slyke and Roger Wets. L-shaped linear programs with applications to optimal control and stochastic programming. *SIAM Journal on Applied Mathematics*, 17(4):638–663, 1969.

[55] R.M. Van Slyke and Roger Wets. L-shaped linear programs with applications to optimal control and stochastic programming. *SIAM Journal on Applied Mathematics*, 17(4):638–663, 1969.

[56] Jean-Paul Watson and David L Woodruff. Progressive hedging innovations for a class of stochastic mixed-integer resource allocation problems. *Computational Management Science*, 8(4):355–370, 2011.

[57] Jean-Paul Watson, David L Woodruff, and William E Hart. Pysp: modeling and solving stochastic programs in python. *Mathematical Programming Computation*, 4(2):109–149, 2012.

[58] EJ Yellig and GT Mackulak. Robust deterministic scheduling in stochastic environments: The method of capacity hedge points. *International Journal of Production Research*, 35(2):369–379, 1997.

[59] Yang Yuan and Suvrajeet Sen. Enhanced cut generation methods for decomposition-based branch and cut for two-stage stochastic mixed-integer programs. *INFORMS Journal on Computing*, 21(3):480–487, 2009.